

Ambulance Agents

How to rescue civilians

- Find it and move to its position
- Get it out `rescue(id)`
- Load it: `load(id)`
- Move to a refuge, if loaded
- Deliver: `unload()`, if at refuge
- Look at `SampleAmbulanceTeam!`

Civilian's properties

- Buriedness: how long it takes one Ambulance to unbury (B/N for N Amb.)
- Damage: Reduction of hp for this round, this value might increase.
- Damage = 0, if at refuge
- Hp: Health as in computer games, 0 = dead.

Coalition formation

- Partition ambulance teams into coalitions
- Evaluate coalitions and chose the one which rescues most civilians
- Whenever something changes (civilian rescued, civilian death, etc.): reevalute
- Optimal evaluation would consider all possible assignment sequences of civilians to the coalitions

Heuristic approach

- We propose the following simple deterministic algorithm to choose assignment sequences
- Each coalition is assigned a sequence (init empty) and a time when they are free to act (init current timestep/0)
- Consider only civilians that are actually buried

Evaluation

- Sort civilians by least buriedness first (ties broken by lowest id)
- In this order add civilians to end of sequence of a coalition
- Which coalition is determined by two things
 - The coalition needs to be able to save the civ
 - Least #agents in coalition wins (ties: lowest agent id in coalition)

Evaluation (2)

- After assignment: add the time to rescue to the free to act time
- If no coalition can rescue a civ, don't assign it
- If at the end a coalition has not been assigned any civs, assign the one with least buriedness, from unassigned civs
- If there are not unassigned ones, use least buriedness from the assigned ones

Simulated rescue

- To determine if a civ can be rescued „simulate“ the rescue
 - Start at free-to-act time
 - Add time to get to civ (max time from all agents)
 - Add unbury time ($\text{buriedness}/\#\text{agents}$)
 - Compare to life time